Title:	Vision Guided Automation and Assistance
Author(s):	Schloen, John Robert
Intended for:	2020 LANL Student Symposium
Issued:	2020-07-27

Disclaimer:
Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Vision Guided Automation and Assistance

By: Robert Schloen

Group: E-3

Mentor: Dr. Beth Boardman

EERE Student

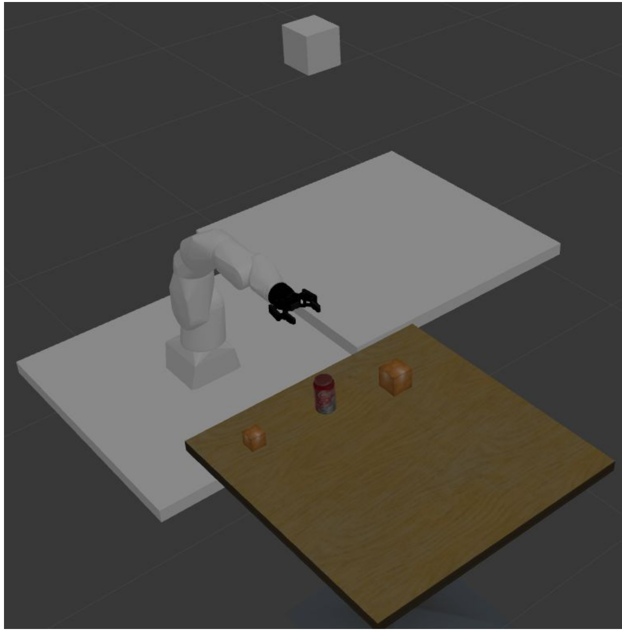
Managed by Triad National Security, LLC for the U.S. Department of Energy's NNSA

Project Goal

- Develop a vision pipeline for Locating and Recognizing objects in a variable workspace for automatic manipulation
  - Depth camera used to visualize objects in space
  - Process point cloud to isolate individual objects
    - Estimate the pose of each object in workspace
  - Apply deep learning to point clouds for recognition
- Human Robot Interaction:
  - Robot uses vision pipeline to assist user in manipulation of items in workspace
  - Example: Human requests robot to move object with visual cue (hand gesture, moveable markers/tags) in workspace

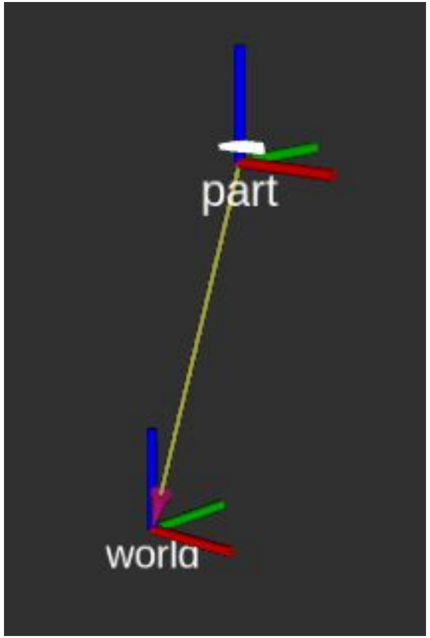
Simulation Environment

- Manipulator
  - Motoman SIA5D Robotic arm with Robotiq 85mm gripper
- Environment
  - Simulated glovebox environment with work surfaces and representative objects
- Vision
  - Simulated depth camera to visualize workspace
  - Angle of camera was set to top down to ensure recognition and distinguishability of objects
  - Fixed location and orientation avoids potential security risks
- Cartesian Path planning
  - Helps to avoid collisions with workspace



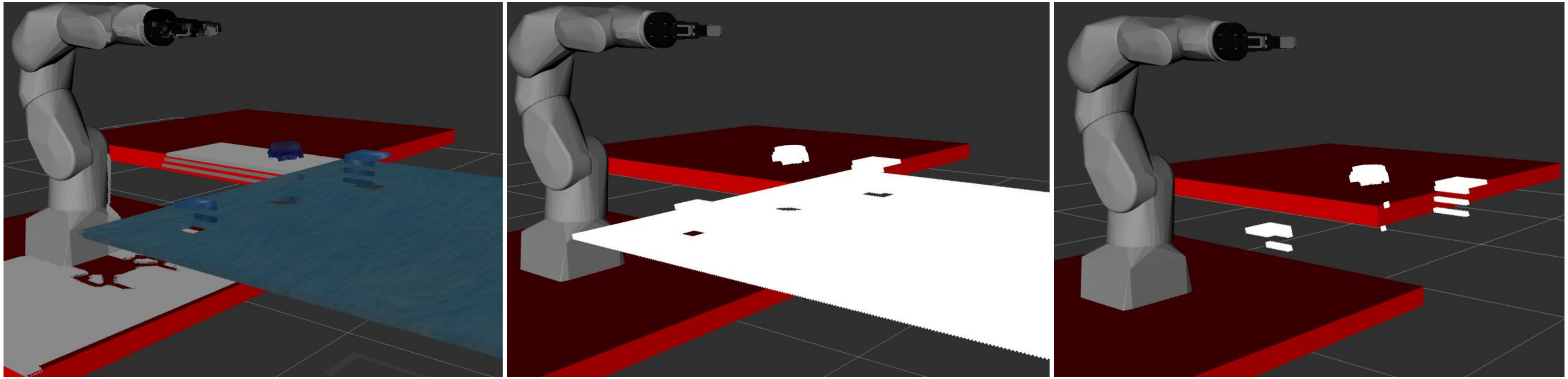
Vision Pipeline

1. Localization
   a. Process point cloud to isolate objects in workspace
   b. Identifying the pose of an object within the workspace
   c. Directs the robot arm in how to approach and grasp the object
2. Recognition
   a. Use Deep Learning to classify objects
   b. Identifies any objects within workspace from known set of objects
   c. Allows robot to distinguish objects and process them separately



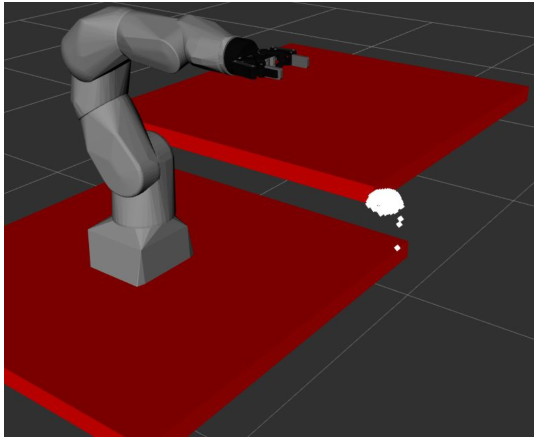
Object Localization

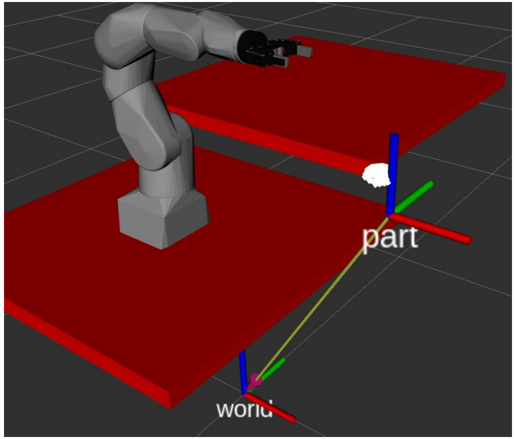
- Pass Through Filter
  - Filter out points that are outside a bounding box set around the relevant workspace
- Plane Segmentation Filter
  - Filter out planes from the point cloud
  - Helps isolate objects by removing flat surfaces such as the floor and the tabletop the objects are resting on

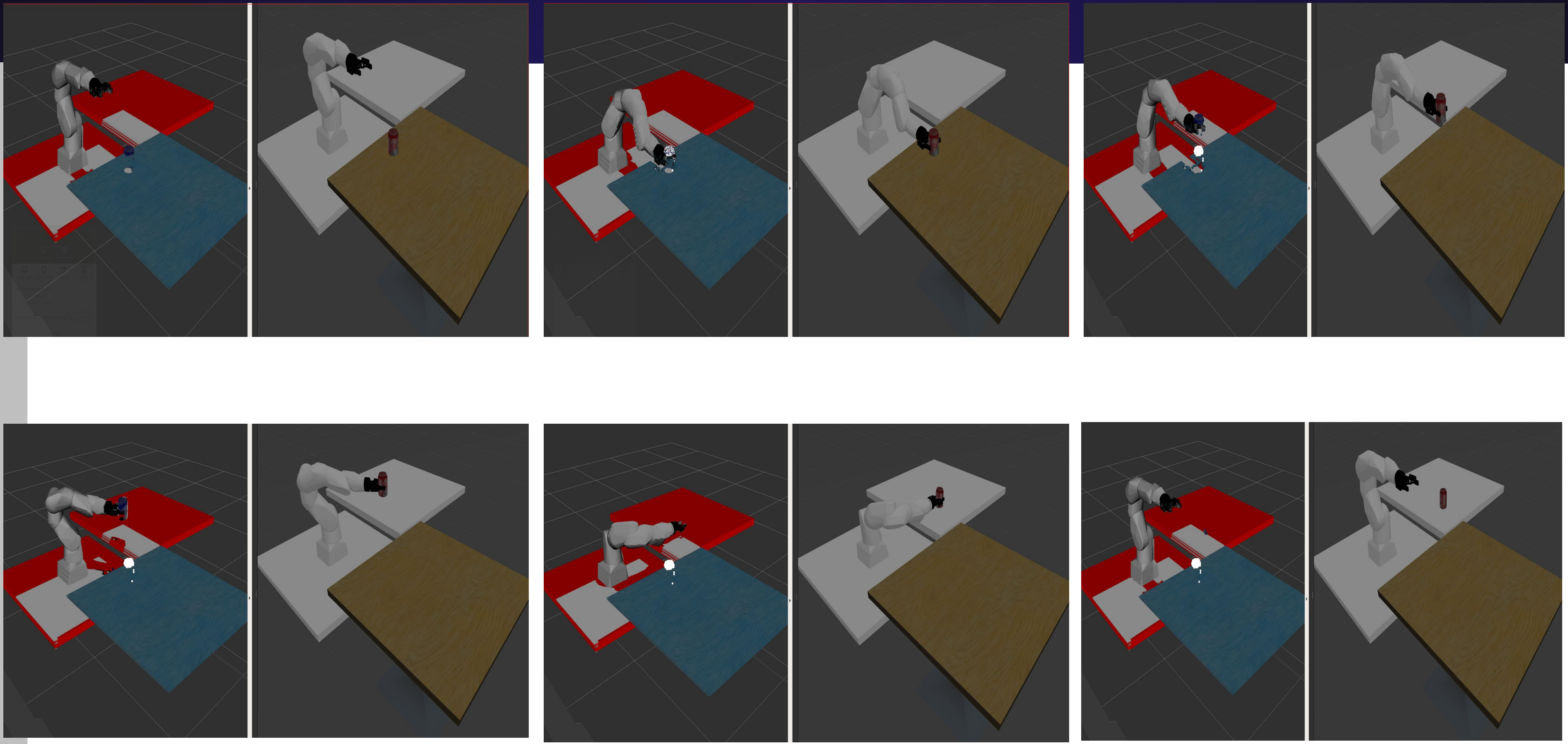


Object Localization

- Euclidean Cluster Extraction
  - Group points into clusters that can represent individual objects or part of an object
  - Isolates multiple objects on surface into individual clouds
- Pose Estimation
  - For any isolated cluster, estimate centroid of cloud

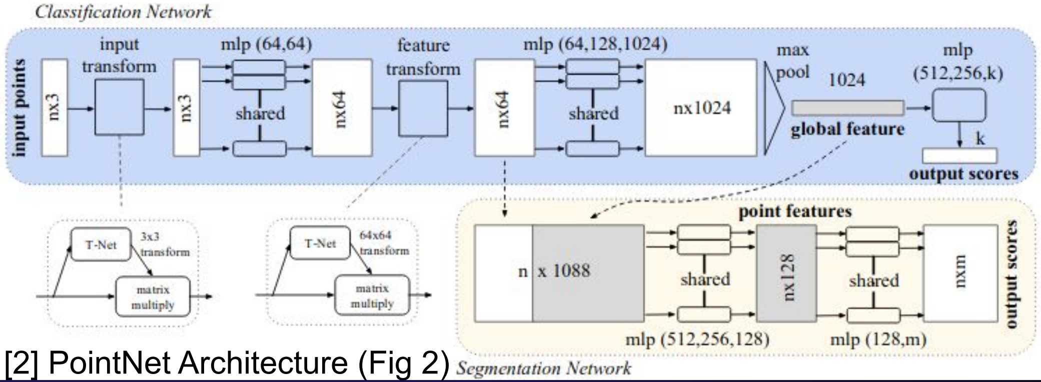






Object Recognition

- Deep neural network (NN) used to classify objects
- Challenges of using depth clouds in neural networks [1]
  - Unstructured
  - Unordered
  - Sparse
- NN Architecture for point clouds
  - Based on PointNet Architecture [2]
  - First transforms input, then passes it through series of convolution, normalization, pooling, and fully connected layers

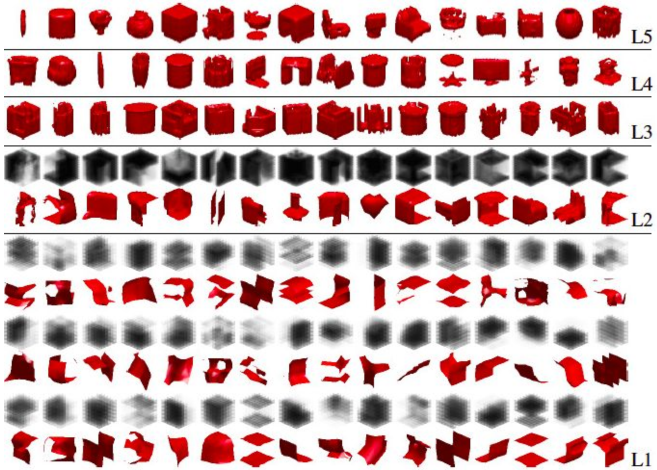


[2] PointNet Architecture (Fig 2)

Object Recognition

- Point Cloud Dataset
  - Combination of existing online datasets and dataset generated from simulation
  - Examples of existing datasets include ShapeNet [3] and ModelNet [4]
- Transfer learning
  - Can retain relationships found from the larger dataset and adjust to fit new data better

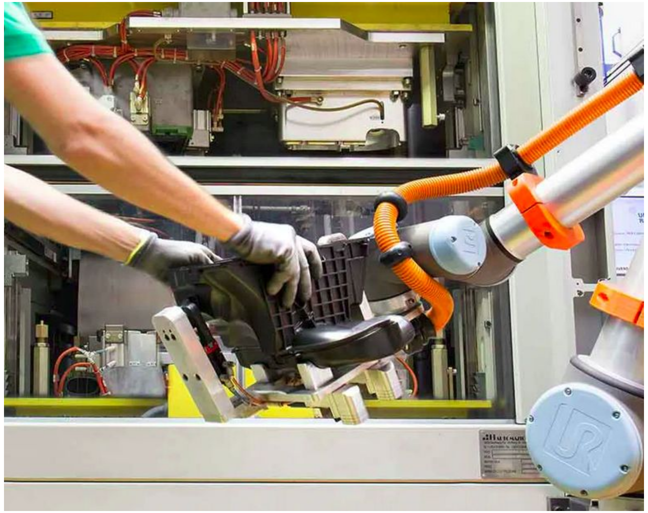
1. Train model on larger dataset, usually a large existing dataset
2. Then retraining model with smaller new dataset for a more specific application



[3] ModelNet Fig. 2

Future work

- Cobotics in glove box setting
    - Aids human user in completing tasks
    - User gives robot visual cues that trigger certain actions
    - Robot uses vision pipeline to identify intended object and performs appropriate actions



References

(Temporary until proper citations added)

[1] S. A. Bello, S. Yu, C. Wang, J. M. Adam, and J. Li, “Review: Deep Learning on 3D Point Clouds,” *Remote Sensing*, vol. 12, no. 11, p. 1729, 2020.

[2] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation,” *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[3] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, “3D ShapeNets: A deep representation for volumetric shapes,” *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[4] A. X. Chang, T. A. Funkhouser, et al. “Shapenet: An information-rich 3d model repository”, *CoRR* abs/1512.03012, 2015.

Robert Schloen

Script for LANL Symposium Presentation

Slide 0: Title Slide

My name is Robert Schloen and today I will be presenting my work on vision guided automation and assistance.

Slide 1: Project Goals

The goal of my project is to develop a vision pipeline that can locate and recognize objects in a variable workspace for use in automated and assistive tasks. For automated pick and place tasks, the placement of the object before being picked up must be very precise. The addition of vision reduces the need for such precision since the position and orientation of the object can be estimated from a point cloud generated from a depth camera. Deep learning can be used to recognize different types of objects, allowing the robot to be used for more complex tasks.

In assistive tasks, where the robot is interacting with a human, vision allows for more flexible interaction, rather than having a few preset actions the robot will execute. For example with vision, the user could use a visual cue to request the robot move an arbitrarily placed tool to a different arbitrary part of the workspace. Without vision, the start and end position of the tool would be limited to preset positions, and another interface would be required to activate the robot.

Slide 2: Simulation Environment

Due to the remote nature of my internship, the project is done entirely in simulation using the simulation and visualization software Gazebo and RVIZ, through the robot operating system, or ROS. The simulation environment was set up to be representative of a glovebox environment found in the lab. For the robotic manipulator, existing robotic systems in the lab were simulated: the Motoman SIA5D arm with a Robotiq 85mm gripper attached as the end effector. Surfaces were added to hold the objects being manipulated. The objects were selected to be representative of the shape of objects found in the glovebox. At the moment, the objects consist of simple shapes like cylinders and cubes, but more complex shapes will be added as the work progresses. For the vision component of the project, a simple depth camera was simulated, which is able to generate point clouds of the simulated environment. The camera is fixed above the workspace with an overhead view, which prevents any object from occluding another, while still having all objects recognizable and distinguishable. Additionally, the fixed location ensures that anything that shouldn't be in view is not, reducing the potential for security risks. When simulating a real life environment, it is important to be aware of possible collisions. To reduce the risk of the arm colliding with surfaces, cartesian path planning, which plans linear trajectories through waypoints, is used when interacting with the objects on the surfaces.

Slide 3: Vision Pipeline

The vision pipeline is composed of two main components: localization and recognition. To locate the object in space, the point cloud from the depth camera is processed and the pose of the objects are found. An example of this can be seen in the image, where the pose of a cube is estimated with respect to the world frame. This information is used to direct the manipulator in how to approach and grasp the object.

Deep learning can be used to classify objects in the workspace. The ability to recognize different objects allows them to be handled appropriately for automation tasks. This increases the complexity the tasks can have. In assistive tasks, recognition can be used to help determine the intent of the user through visual cues, and to identify the correct object to manipulate.

Slide 4: Object Localization (1)

Locating an object from a point cloud means you have to isolate the points that make up the object from the rest of the points in the cloud. An example of the original point cloud is shown at the bottom of the slide. When processing the point cloud, first a pass through filter is applied which removes any points that fall outside a given bounding box. In this case, the surface where we are looking for objects is fixed, so the bounding box is placed to only contain that surface. The remaining points are shown in the second image. Next, any remaining large flat surfaces, identified using plane segmentation, are filtered out. This leaves only the points that correspond to the objects.

Slide 5: Object Localization (2)

At this point, the points for all the objects have been isolated, but the points are not yet associated with just one object. To distinguish the individual objects, euclidean cluster extraction is used to group points that are close together into separate clusters. This allows us to isolate the points for just one object, as shown in the first picture where the cubes are filtered out leaving just the cylinder. Finally, using the point cloud for just the object of interest, in this case the cylinder, the centroid of the cloud is estimated, giving the pose of the object.

Slide 6: Demo

Now I'll walk through a demonstration of using the localization component of the pipeline with the robot. First, the robot arm is shown in a home position with an object on the surface. The left half of the image shows the gazebo simulation environment, and the right half is the visualization in RVIZ, where the point cloud can be visualized. From the point cloud, the location of the can has been found using the steps previously described. The robot plans a path from its home position to the position of the can. In the next image, the robot has executed that path and grasps the can. It plans the path to the drop off points on the next table, and begins moving the can. You can now more clearly see the white point cloud of the can that was used to estimate the can's location. The robot continues through its planned path and places the can on the other surface. Finally, the arm is returned to the home position where it waits for the next object.

Slide 7: Object Recognition (1)

For the recognition part of the vision pipeline, I will be using Deep learning. Deep neural networks have become more popular in recent years and are a powerful tool for tasks such as image recognition. Deep neural networks consist of a series of layers through which an input is passed, returning an output. In the case of a classification task such as the one I am working on, the output would correspond to the type of object. There are, however, some challenges with passing a point cloud through a neural net. Point clouds are unstructured since they don't fall on a grid like 2D images, which are a grid of pixels. Point clouds are also unordered; where you can list the points in any order and get the same result. This is unlike 2D images where if the order of the pixels is changed the image changes. They can also be sparse and irregularly distributed where you can have more points in one area of a cloud than another. With these challenges in mind, I will be using a neural network based on the PointNet architecture. PointNet can be considered the foundation for most neural nets using raw point clouds. The neural network for this project will first transform the inputs and pass them through a series of layers, including convolution, pooling, and normalization, and output scores for each object class.

Slide 8: Object Recognition (2)

Deep neural networks require a lot of data to train on with enough variation that the trained model is robust to noise and generalizable to new data it hasn't seen before. Since it can be challenging to build a dataset from scratch, it can be useful to use an existing dataset that has similar classes of objects. For this reason I will be using an existing online dataset, such as ModelNet, as well as a smaller dataset generated from my simulated environment. The reason the larger dataset is helpful, even though it is less specific to the task, is due to the idea of transfer learning. Transfer learning is the idea that the neural net can learn more general relationships from the larger and less specific dataset, which can then be transferred to the smaller dataset. The general relationships are kept while the specific relationships are adjusted to better fit the more relevant data. Common transfer learning techniques have two main steps. First, the neural net is trained on the large existing dataset. Then the general relationships you want to preserve, which are the trained weights and parameters of each layer, are frozen so they won't change. Finally, the network is retrained with the smaller specialized dataset, except for the frozen parameters which have preserved the general relationships from the larger model. As a result, we have trained a more generalizable and robust network with the small dataset than if the network had been trained on just the small dataset.

Slide 9: Future Work

Another area that would greatly benefit from the vision pipeline is cobotics, or collaborative robotics. Cobotics in a glovebox setting, where the robot is working alongside and assisting a human user, would require an additional visual interface for communicating with the robot. The user's movement can be limited in a glovebox, so possible interfaces could include recognizing basic hand signals, or easy to move markers or tags. The markers could be used to tell the robot what to manipulate and where to move objects. While it could not be explored in depth this summer, cobotics is another area which, like automated tasks, would have improved flexibility and efficiency with the addition of robot vision.